DOCUMENT RESUME

ED 371 746 IR 055 066

AUTHOR Blazek, Daniel
TITLE Reclassifying Florida State Documents: Down to the Nitty Gritty.
PUB DATE [94]
NOTE 12p.
PUB TYPE Reports - Descriptive (141)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS Cataloging; *Classification; College Libraries; Computer Software; Documentation; *Government Publications; Higher Education; Indexes; *Library Collections; Preservation; Programming; Recordkeeping; Records Management; *State Government
IDENTIFIERS Call Numbers; dBase; Florida Atlantic University; Keywords; Labeling (Of Objects); *Reclassification; University of Miami FL

ABSTRACT

The University of Miami recently reclassified their "State of Florida" documents according to the classification system invented by Florida Atlantic University (FAU) in 1966. Using only in-house resources, the Government Documents Department of the University of Miami generated labels with printed call numbers for over 16,000 documents. This article describes the steps in the project down to the level of detailed software programming. The FAU system is noted for its excellent Keyword-In-Context Indexes, characterized by frequent updating and maintenance. For every piece in the collection, a call number had to be found or created. Once call numbers were identified, computer labels were generated and printed using Nutshell and dBase software. Two figures illustrate the conversion slip and a layout screen for data entry. (SLD)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☐ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

ED 371 746

Reclassifying Florida State Documents: Down to the Nitty Gritty

by Daniel Blazek

University of Miami
6860 SW 45th Lane #2
Miami, FL 33155

IR055066

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Daniel Blazek

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

BEST COPY AVAILABLE

IR

Reclassifying Florida State Documents: Down to the Nitty Gritty

by Daniel Blazek

The University of Miami recently reclassed their state of Florida documents to the FAU classification system. Using only in-house resources, the Government Documents Department generated labels with printed call numbers for over 16,000 documents, enabling easier access and faster processing. This practical article details the steps of the project down to the level of detailed software programming.

Although State of Florida Documents have been distributed by the State Library since 1967,[1] state documents have never had an official classification system. Unlike other state depository systems, it has never been the state library's mandate to provide call numbers for documents and it has been left to individual libraries to decide how to arrange their collections.

In 1966, however, Florida Atlantic University (FAU) invented a classification system for state documents which in time proved to be the most widely used class system among state depositories.[2] Termed "archival" by Castonguay,[3] the FAU classification system is agency based like GPO's Sudoc system and continues to be popularized largely by their excellent Keyword-In-Context Indexes.[4] These FAU Indexes provide an avenue to document titles and supply a corresponding FAU call number, making them one of the best access tools available for state documents. Using FAU's class system, it is possible to go directly from the Keyword Index to a document on the shelf.

Largely because the FAU class system was and still is unofficial, the University of Miami had maintained its own local numbering system for Florida documents even before 1967, the year in which the state depository system was established. However, as time went on, it proved beneficial to reclassify Miami's Florida documents to the FAU system. FAU's Keyword Indexes continue to be maintained and updated by FAU librarians[5] and are distributed by the state library to state depositories. Monthly updates of call numbers are also provided by FAU direct to depositories using their system, making the FAU class system the closest thing to an official state documents classification scheme. Although Miami's local class system was similar to FAU's in that it was arranged by agency, Miami's system was overly-demanding of professional time to apply it to monthly depository shipments, and seemed to replicate the efforts of FAU without the benefit of being able to use the call numbers provided in the FAU Indexes.

In order to save money and time in the long run, reclassification was necessary. No longer would hours be spent classifying documents in a local scheme. FAU's monthly updates would provide the call number. Once reclassification was completed, patrons and staff would be able to access

IR055066

documents directly from the FAU Indexes, eliminating the need to consult the shelflist for a local call number. This article summarizes UM's experience with state document reclassification down to the details of computer nitty gritty, which was often learned by trial and error. For more detailed reasons to undertake a reclassification project, consult Dean's, "Classification in an Automated Environment".[6]

The Ground Floor.

Because there were no vendors who supplied pre-printed call numbers for Florida documents, reclassification had to be done in house. Taking on such a project with limited resources can be done, but it is important to use computers wherever possible to reduce time spent in repetitive tasks. Also essential is a little software know-how which will prevent computers from becoming time sponges, frustrating you with unforseen problems. Since reclassification is such a large and fundamental change to any collection, limited resources require ingenuity and adaptability.

Reclassification involved finding a new call number for each document, keying the new call number into Nutshell software, and then generating a label through Dbase to affix to the document. Nutshell software was chosen for its easy data entry screen and Dbase was used for label generation.

The following steps describe UM's experience dealing with reclassification. Specific software commands are detailed for practical purposes and Dbase label generation is fully explored. 23 years of state documents amounted to over 4000 brief records to be keyed. Because of a large number of serial titles, this totaled to over 16,000 individual pieces to be relabeled. Hopefully some of the details will help other libraries who are contemplating a similar project.

Stairwell into Reclassification

1) Finding the New Number.

Documents were taken off the shelf in old call number order and titles were looked up one by one in FAU's Florida Keyword Indexes. The Indexes of FAU functioned as the source for new call numbers, which were accepted as is. Occasionally, a title was not in the Keyword Index, so the document was put aside and a new call number was invented at a later date.

Once the number was found, data used to identify the document was written on brief number conversion slips, which numbered eight to a sheet. (Figure 1). The following fields proved to be useful for identification and relabeling: title, old call number, new call number, and the number of labels to be printed. However, only the new call number and number of labels are essential for label production.

It was soon discovered that the entries for the new and old call numbers had to be broken down into three separate fields to facilitate sorting once the data was entered into Nutshell

software. In Nutshell, each field is defined at the time of creation as either text, number, date or other calculation field. Since call numbers are composed of both text and numbers, they had to be subdivided in order for a correct sort to occur. In the case of Florida documents, both the new and old call numbers were composed largely of letters, followed by numbers and then more letters. So in like fashion, the call numbers were written down and entered in three fields of similar arrangement: text, number, text.

Also on the slips, but not often used, were fields to indicate if an item was bound or to be weeded. Items which were bound required manual hot-pen renumbering so a label was not printed. Likewise, material which was judged not worth keeping (indicated by a "W?"--for Weed?) was disposed of and no label was created. In either case, however, a slip was filled out and the number of labels to be printed was marked "zero".

Documents which had their data analyzed on slips were kept in old call number order on holding shelves away from the public. These would await data entry and the printing of labels.

2) Entering Data into Nutshell.

The new call number as well as the other elements described on the data sheets were entered by student workers into a file manager called Nutshell[7], an application which proved fool-proof for data entry. Although Nutshell is not programmable in the strict sense, multiple layouts can be easily created for a single file. Nutshell is user-friendly and the tutorial walks you through the core of its operation. Its main drawback, however, is that it only allows one label to be printed per record. In order to print multiple labels from an entry (in the case of serials), the data must be transferred to another application, such as Dbase. For ease-of-use at the front end, however, Nutshell was exemplary and this was important since students performed the bulk of data entry.

The layout screen for data entry was created to mirror the fields on the data sheets. (Figure 2). The only difference was a date of entry field was added so that portions of the records could be identified and printed in batches. This was important considering the size of the project.

There was very little supervision required for Nutshell data entry aside from showing students the basic keys. The [Tab] key is used to jump from field to field and [F10] is used to go to the next record. There is no save key as everything in Nutshell is saved automatically when entered.

3) Printing Labels (Descent into Automation)

In order to print labels, a three-part process was involved: Nutshell data transfer into ASCII, Dbase file structure creation with ASCII data merge, and Dbase programming for label generation. Although this may sound complex, essentially what was done was transfer the data entered in Nutshell to Dbase so that multiple labels could be printed from the same entry. Dbase is not quite as simple to use as Nutshell, but it is widely used and

fully programmable.

A) Nutshell Data Transfer to ASCII File.

ASCII files are the esperanto of software, allowing transfers of data between different programming languages. Creating an ASCII file in Nutshell is very easily done, although some preparation of records was necessary to target the output of particular records.

Labels were printed in batches of roughly 750 at a time by first identifying records through the "date of input" field in Nutshell. Since each record had chronological input, a batch of records could be distinguished by using the Find command [Alt-F] in the date field and limiting the output of records. Once the batch of records was found, only those records would be used in a data transfer.

Also, in Nutshell only those fields which are on the current layout are transferred in a data export. This was used to advantage to create a layout that contained only the new call number fields and the number of labels field. These were the only fields necessary for label generation. (It is important to note that the date field must be searched first before switching to a layout that has only the fields mentioned above. It is impossible to search a field that is not present in the current layout.)

With the batch of records identified by date and the current layout containing only the necessary fields, an ASCII file named MAIN.ASC was created by using Output (Alt-O) and export (Alt-E). Remember to note the order of the fields, because it is important that they are not confused during transfer. Make sure the transferred ASCII file has an extension, because Dbase will look for a file with a .dbf extension during the file merge if you do not assign it one. After ASCII export, Dbase is now used for label generation.

B) Dbase File Structure Creation.

Dbase's file command mode is a popular way to set up a file structure to input data. A file called MAIN.DBF was set up using DBase's Create function. Once the file's fields were established, two simple commands transferred the data in the ASCII file (MAIN.ASC) to the Dbase file structure (MAIN.DBF).

When creating a structure, Dbase will prompt you for field names, the type of field (character, numeric, etc.) and the field width. For MAIN.DBF, field names were borrowed verbatim from the field names used in Nutshell and entered in the same order. This kept things as uncomplicated as possible.

create main.dbf

Field Name	Type	Width
newcall1	character	10
newcall2	character	10
newcall3	character	10
labels	character	10

The type of field was always designated as a character field, even though in Nutshell the fields for "labels" and "newcall2" were originally created as numeric fields. This was done mainly to keep all the commands in the following Dbase print program (LABELS.PRG) consistent. The field width was given a generous ten character spaces each to ensure long numbers would not be clipped.

After the file structure was created and saved, Dbase will ask if you want to enter records. Respond no, and then type the following commands:

```
use MAIN.DBF
append from MAIN.ASC type delimited
```

"Use" ensures that the correct file is opened. "Append from" enters the contents of the ASCII file [MAIN.ASC] into the newly created dbase file [MAIN.DBF]. "Type delimited" simply means that the data is in ASCII format. Dbase will add the data and display the number of records. The next step is programming the print program.

C. DBASE Print Program

Most people who use Dbase use it in the Assist mode or the "dot prompt" mode where commands are immediately evaluated and results displayed. But Dbase is also a complete programming language that can be used to run multiple commands for more complex operations.[8] To create a program in Dbase, type "Modi Comm" (or Modify Command) at the dot prompt. It will prompt you for a file name. If no extension is specified, Dbase will automatically give it an extension of .PRG. The following brief program called LABELS.PRG was used to print out the correct number of labels for each record from the file MAIN.DBF. (Brief explanations are in brackets:)

LABELS.PRG

```
set talk off   [sets terminal not to receive feedback]
Counter = 0    [counter is a value used for keeping track of the
               number of labels to be printed in do loop]
use main.dbf   [uses file main.dbf]
go top         [starts at the top]
do while .not. eof()
     do while val(labels) > counter
               [labels is a field in main.dbf, which is the
               number of labels to be printed for each call
               number. Val converts a text field to a its numeric
               designation.]

     ?   trim(newcall1)   +   trim(newcall2)   +   trim(newcall3)
               [Trim erases extra spaces from the three fields of
               the new call number. ? sends the data to
```

```
                the screen or printer.]

     counter = counter + 1  [counter is incremented]
     ? ""        [5 blank lines are displayed/printed]
     ? ""
     ? ""
     ? ""
     ? ""
     enddo       [Call numbers with more than one label are
                 returned through do loop. When counter equals
                 labels, loop stops]
skip 1           [goes to next record]
counter = 0      [resets counter to 0]
enddo
```

Once the program is entered and saved, the command "Do" followed by the program name runs the program. A good idea is a test run which outputs data only to the terminal screen. Once everything appears okay, the command "set print on" is entered, sending output to the printer. Avery self-adhesive labels were fed through a standard Epson printer and a small batch was printed before turning the printer off-line. This was done to center the call number on the label. After adjusting the line spacing, resume printing by turning the printer on-line.

4) Attaching Labels.

Once a batch of labels was generated, the number conversion slips which had been filled out for data entry served as guides for identifying documents when attaching labels. Lists could have been generated from the information entered into Nutshell, but the conversion slips worked just as well. Since the documents were analyzed in old call number order, the data sheets were generally in old call number order, as were the printed labels. Student-workers matched the new call number on the data sheets with the number on the label and affixed the labels over the old call number on the document.

For various reasons, it was decided not to spend time updating the manual shelflist. Labels could have been printed for the cards, but since the library had acquired a new integrated system, it was felt the time would be better spent pursuing retrospective conversion. This meant that once the label was attached, the document could be placed on public shelving. Some serials needed the addition of a date or series number at the end of the call number, so this was done manually.

Conclusion

Reclassification is time consuming, no way around it. Some tasks such as determining the correct new call number to be assigned to a document cannot be done by a computer or a student worker; professional judgement is required. For every piece in a given collection, a new call number must be found or created.

Once the new call number is identified, however, computers are ideal for printing labels, generating lists and greatly reduce time spent renumbering. Entering data into software applications may also be time consuming, but this can be done by non-professional staff. Handsome labels can be generated both for the documents themselves, and their respective shelflist cards, if still in use.

The entire reclassification project lasted about a year, with various levels of involvement dedicated at different times. Finding the new call number was the most time consuming aspect of the project as months were spent filling out number conversion slips. Although data entry did not involve professional labor, it too stretched out over a period of months.

Separate Nutshell files were created for subgroups of the collection, including vertical file material, weeded material and documents for which FAU call numbers could not located. Data entry and reclassification for these exceptions took place after the bulk of the collection, so as not to absorb the momentum of the project. It was extremely important not to get bogged down in the nitty gritty.

One last word of note, if you do not need a list of titles generated you might save some time by not entering titles and old call numbers into your computer files. Write them down to serve as guides for label attachment, but they are not neccesary for label production. Only the new call number and number of labels to be printed are necessary to print labels. For the University of Miami, the lists of titles were used in the reclassification of the library's second collection of state documents, which are housed in the Archives department, and also in the production of a small Florida documents keyword index supplement which was comprised of documents not listed in FAU's Indexes.[9]

1. "An Act Relating to the State Library and Historical Commission..." (Chapter 67-223, 1967) Laws of Florida, p. 467.

2. "Survey of State Documents Depository Libraries, June 1993"; State Library of Florida. Unpublished materials provided at the biennial State Depository Workshop of October 1, 1993, St. Petersburg, Fla.

3. Castonguay, Russell. A Comparative Guide to Classification schemes for Local Government Documents Collections. Westport, CT: Greenwood Press, 1984. p.95.

4. A Keyword-in-Context Index to Florida Public Documents in the Florida Atlantic University Library. v.1, 1969-1979. Edited by Elaine K. Kelly. v.2, 1980-. Edited by Margaret Stone Walker. Tallahassee, Fl: State Library of Florida, 1980-.

5. Kudos to Margaret Stone Walker, Documents Librarian and her staff.

6. Barbara Dean, "Reclassification in an Automated Environment," Cataloging and Classification Quarterly, 5, (Winter 1984): p.1-3.

7. There is some doubt as to whether or not the producers of Nutshell are still in operation. The author could not locate Nashoba Systems Inc. of Concord, Massachusetts.

8. Douglas Hergert, dBase III: The Microsoft Desktop Dictionary and Cross-Reference Guide. Bellevue, Wash.: Microsoft Press, 1985. p.264.

9. Florida Keyword Index Supplement, Created by Daniel Blazek and Dennis Yancey. Miami: Univ. of Miami, Government Documents Department, 1992.

(Fig.1)

FLORIDA DOCUMENTS NUMBER CONVERSION SLIP

New Number AD.B 2 :I59/975

Title Interim Report on the Special Project to Prevent Eutrophication of Lake Okeechobee

Old Number ADM. 61 /P43:04

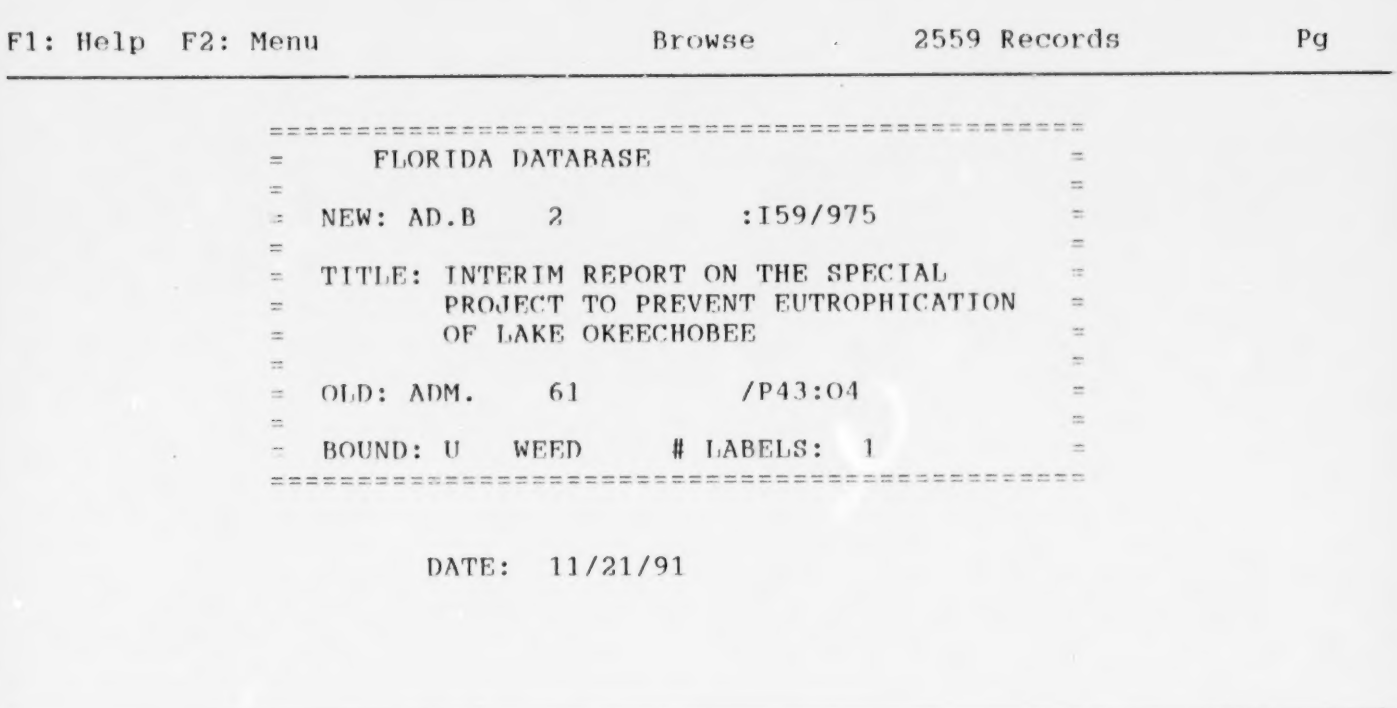
W? Bound or Unbound

Number of Labels 1

Entered into Computer _____

BEST COPY AVAILABLE

(Fig.2)


F1: Help F2: Menu
Browse
2559 Records
Pg
FLORIDA DATABASE
NEW: AD.B 2 :I59/975
TITLE: INTERIM REPORT ON THE SPECIAL
PROJECT TO PREVENT EUTROPHICATION
OF LAKE OKEECHOBEE
OLD: ADM. 61 /P43:04
BOUND: U WEED # LABELS: 1
DATE: 11/21/91
_ Find _ Add _ Modify _ Remove _ Sort _ Output _ Layout _ Define